REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
3/1/2012

2. REPORT TYPE
Final Technical

3. DATES COVERED *(From - To)*
4/1/2009 - 11/3/2011

4. TITLE AND SUBTITLE
NON-INTRUSIVE MEDIA FORENSICS FRAMEWORK

5a. CONTRACT NUMBER

5b. GRANT NUMBER
FA9550-09-1-0179

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Prof. K. J. Ray Liu

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Department of Electrical and Computer Engineering
University of Maryland, College Park, MD

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Office of Scientific Research
Suite 325, Room 3112
875 Randolph Street
Arlington, VA 22203-1768

10. SPONSOR/MONITOR'S ACRONYM(S)
AFOSR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-OSR-VA-TR-2012-0986

12. DISTRIBUTION / AVAILABILITY STATEMENT
Distribution A - Approved for Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT

In recent years, the availability of high-quality digital cameras and audio recording devices coupled with the rise of the Internet as a means of information delivery has cause digital content to become prevalent throughout society. Many governmental, legal, scientific, and news media organizations rely on digital multimedia content to make critical decisions or to use as evidence of specific events. This proves to be problematic, as the rise of digital media has coincided with the widespread availability of digital editing software. At present, a forger can easily manipulate digital content such as images or video to create perceptually realistic forgeries. To avoid both embarrassment and legal ramifications, many of these organizations now desire some means of identifying alterations to digital multimedia content and verifying its authenticity. As a result, the field of digital multimedia forensics has been born.

Digital multimedia forensics is the study and development of techniques to determine

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Robert Herklotz
a. REPORT U	**b. ABSTRACT** U	**c. THIS PAGE** U	UU	19	**19b. TELEPHONE NUMBER** *(include area code)* 703-696-6565

Final Technical Report
Grant Title: NON-INTRUSIVE MEDIA FORENSICS FRAMEWORK
Grant Number: FA9550-09-1-0179

Principle Investigator: Prof. K. J. Ray Liu

Department of Electrical and Computer Engineering
University of Maryland, College Park, MD

I. Introduction

In recent years, the availability of high-quality digital cameras and audio recording devices coupled with the rise of the Internet as a means of information delivery has cause digital content to become prevalent throughout society. Many governmental, legal, scientific, and news media organizations rely on digital multimedia content to make critical decisions or to use as evidence of specific events. This proves to be problematic, as the rise of digital media has coincided with the widespread availability of digital editing software. At present, a forger can easily manipulate digital content such as images or video to create perceptually realistic forgeries. To avoid both embarrassment and legal ramifications, many of these organizations now desire some means of identifying alterations to digital multimedia content and verifying its authenticity. As a result, the field of digital multimedia forensics has been born.

Digital multimedia forensics is the study and development of techniques to determine the authenticity, processing history, and origin of digital multimedia content without relying on any information aside from the digital content itself. In the past, digital watermarking techniques have been proposed as a means to accomplish these tasks. For watermarking techniques to be successful, however, an extrinsic watermark must be inserted into the digital content by a trusted source before any manipulation occurs. This is unrealistic in many scenarios, because the party that captures the digital content can alter it before inserting the watermark. By contrast, digital forensic techniques operate by searching for *intrinsic fingerprints* introduced into digital media by editing operations and the digital capture process itself. Because most signal processing operations leave behind unique intrinsic fingerprints, no universal method of detecting digital forgeries exists. Instead, several forensic tests must be designed to identify the fingerprints of a variety of digital content editing operations. It has been posited that if a large set of forensic methods are developed, it will be difficult for a forger to create a digital forgery capable of fooling all forensic authentication techniques [1].

Though existing digital forensic techniques are capable of detecting several standard digital media manipulations, they do not account for the possibility that *anti-forensic* operations designed to hide traces of manipulation may be applied to digital content. In reality, it is quite possible that a forger with a digital signal processing background may be able to secretly develop anti-forensic operations and use them to create undetectable digital forgeries. To protect against this scenario, it is crucial for researchers to develop and study anti-forensic operations so that vulnerabilities in existing forensic techniques may be known. This will help researchers to know when digital forensic results can be trusted and may assist researchers in the development of improved digital forensic techniques. The study of anti-forensic operations may also lead to the identification of the intrinsic fingerprints of anti-forensic operations and the development of techniques capable of detecting when an anti-forensic operation has been used to hide evidence forgery.

We have developed a wide variety of digital multimedia forensic and anti-forensic techniques. In Section II, we present the our work on the detection of image manipulation using statistical intrinsic fingerprints.

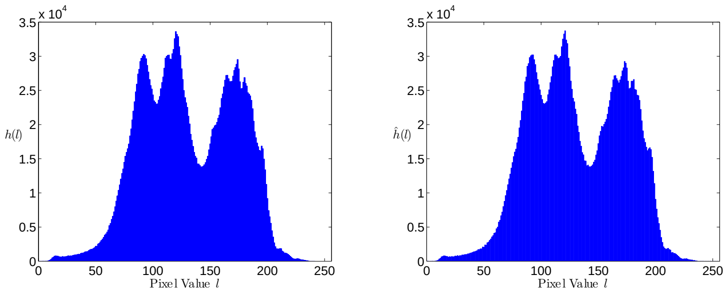


Fig. 1. Left: Histogram of a typical image. Right: Approximation of the histogram at left by sequentially removing then interpolating the value of each histogram entry.

In Section III, we discuss our work on anti-forensically removing compression fingerprints from digital images. We present our work on video frame deletion forensics and anti-forensics in Section IV. Finally, in Section V we discuss our work on evaluating the effectiveness of anti-forensic techniques and analyzing the interplay between forensics and anti-forensics using game theory.

II. Image Forensics via Statistical Intrinsic Fingerprints

In this section, we discuss our forensic work aimed at detecting digital image manipulation. Specifically, we present methods designed to detect general forms globally and locally applied contrast enhancement, and show how the detection of localized contrast enhancement can be used to identify cut-and-paste type image forgeries [2] [3] [4]. We present a technique to jointly estimate the contrast enhancement mapping used to modify an image as well as the images pixel value histogram before contrast enhancement [5]. Additionally, we present a method to detect the global addition of noise to a previously JPEG compressed image by detailing the effect of noise on the fingerprint of a known pixel value mapping applied to the image in question [3] [4]. Each of these techniques identify image manipulation by detecting the presence or absence of the statistical intrinsic fingerprints introduced into an image's histogram by pixel value mappings.

A. System Model

When analyzing a digital image, a histogram $h(l)$ of the color or gray level values l recorded at each pixel can be generated by creating L equally spaced bins which span the range of possible pixel values, then tabulating the number of pixels whose value falls withing the range of each bin. Unless otherwise specified, we will hereafter assume that all gray level values lie in the set $\mathcal{P} = \{0, \ldots, 255\}$, all color values lie in the set $\mathcal{P}^3$, and that all pixel value histograms are calculated using 256 bins so that each bin corresponds to a unique pixel value. After viewing the pixel value histograms of several camera generated images corresponding to a variety of scenes, we have observed that these histograms share common properties. None of the histograms contain sudden zeros or impulsive peaks. Furthermore, individual histogram values do not differ greatly from the histogram's envelope. To unify these properties, which arise due to observational noise [6], sampling effects, and complex lighting environments, we describe pixel value histograms as *interpolatably connected*. We define an interpolatably connected histogram as one where any histogram value $h(l)$ can be approximated by $\hat{h}(l)$, the interpolated value of the histogram at pixel value l calculated using a cubic spline given $h(t)$ for all $t \in \mathcal{P} \setminus l$. The histogram of a typical unaltered image as well as its approximation $\hat{h}$, where each value of $\hat{h}$ has been calculated by removing a particular value from h then interpolating this value using a cubic spline, are shown in Fig. 1. As can be seen in this example, there is very little difference between the image's histogram and its approximation.

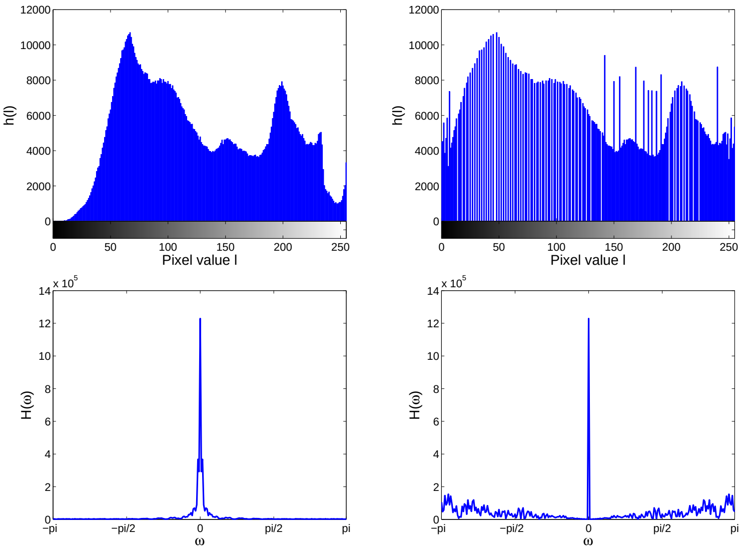


Fig. 2. Pixel value histogram of an unaltered image (top left) and the same image after contrast enhancement has been performed (top right), as well as the magnitude of the DFT of the unaltered image's histogram (bottom left) and the contrast enhanced image's histogram (bottom right).

B. *Detecting Globally Applied Contrast Enhancement*

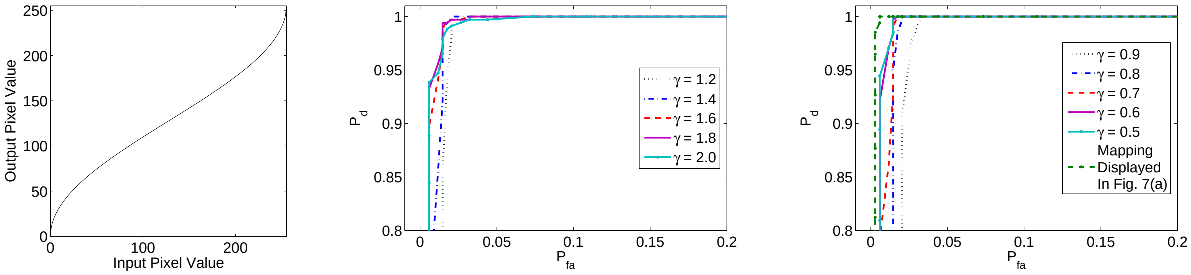
Contrast enhancement is an image processing operation commonly used to compensate for poor lighting conditions. It operates by applying a nonlinear mapping to the pixel values of an image in order to increase their effective dynamic range. A wide variety of contrast enhancement mappings are commonly used, several of which cannot be simply parametrically described. Each of these mappings, however, must necessarily map multiple input pixel values to the same output pixel value. This will introduce impulsive peaks and zeros into the pixel value histogram of a contrast enhanced image, as can be see in Figure 2 which shows the histogram of an image before and after contrast enhancement. These peaks and zeros correspond to the intrinsic fingerprints of contrast enhancement mappings.

We have proposed a technique to detect these contrast enhancement fingerprints using a frequency domain representation of an image's pixel value histogram. Because the pixel value histogram of an unaltered image should be 'smooth', the Fourier transform of that image's histogram should be a strongly low-pass signal. Contrast enhancement fingerprints introduce energy into the high frequency components of an image's pixel value histogram due to their impulsive nature. Both of these phenomena can be observed in the bottom two plots of Figure 2. As a result, we detect contrast enhancement by measuring the strength of the high frequency components of an images pixel value histogram, then comparing this measurement to a predefined threshold.

To test of our contrast enhancement detection technique, we compiled a database of 341 unaltered images consisting of many different subjects and captured under a variety of light conditions. These images were taken with several different cameras and range in size from 1500×1000 pixels to 2592×1944 pixels. The green color layer of each of these images was used to create a set of unaltered grayscale images. We applied the power law transformation defined as

$$m(l) = 255\left(\frac{l}{255}\right)^{\gamma}, \tag{1}$$

to each of these unaltered grayscale images using γ values ranging from 0.5 to 2.0 to create a set of contrast enhanced images. Additionally, we modified each unaltered grayscale image using the nonstandard contrast



(a) (b) (c)

Fig. 3. Contrast enhancement detection ROC curves for images altered by a power law transformation with (b) $2.0 \geq \gamma \geq 1.2$, and (c) $0.5 \geq \gamma \geq 0.9$ as well as the mapping displayed in Fig. 3(a).

enhancement mapping displayed in Fig. 3(a). These images were combined with the unaltered images to create a testing database of 4092 grayscale images. To evaluate the performance of our contrast enhancement detection technique on this testing set, each image was classified as altered or unaltered using a series of decision thresholds. The probabilities of detection P_d and false alarm P_{fa} were determined for a series of decision thresholds by respectively calculating the percent of contrast enhanced images correctly classified and the percent of unaltered images incorrectly classified. These results were used to generate the series of receiver operating characteristic (ROC) curves shown in Figs. 3(b) and (c). For each form of contrast enhancement tested, our detection technique achieved a P_d of 0.99 at a P_{fa} of approximately 0.03 or less.

C. Detecting Locally Applied Contrast Enhancement

Locally applied contrast enhancement can be defined as applying a contrast mapping to a set of contiguous pixels within an image. If the size of this set of pixels is large enough for our pixel value histogram model to remain valid, then when contrast enhancement is performed it will introduce its fingerprint into the histogram of this set's pixel values. In light of this, we have proposed detecting locally applied contrast enhancement by segmenting an image into a set of blocks, then performing contrast enhancement detection on each block. The blockwise detection results can be combined to identify image regions which show signs of contrast enhancement.

In order to determine which block sizes are sufficient to perform reliable detection and examine the effectiveness of the local contrast enhancement detection scheme, we performed the following experiment. Each of the 341 unaltered images from the test database described in Section II-B along with the power law transformed images corresponding to $\gamma = 0.5$ through 0.9 were segmented into square blocks of varying sizes. Each block was then classified as contrast enhanced or unaltered using by our contrast enhancement detection scheme using a variety of different thresholds. False alarm and detection probabilities were determined at each threshold and for every choice of block and were used to generate the set of ROC curves shown in Fig. 4 for each value of γ which was tested. These ROC curves indicate that local contrast enhancement can be reliably detected using testing blocks sized least 100×100 pixels. At a P_{fa} of approximately 5%, a P_d of at least 95% was achieved using 200×200 pixel blocks and a P_d of at least 80% was achieved using 100×100 pixel blocks for each form of contrast enhancement tested.

In some scenarios, locally applied contrast enhancement detection can be used to identify other, more obviously malicious image manipulations such as cut-and-paste forgery. Cut-and-paste image forgery consists of creating a composite image by replacing a contiguous set of pixels in one image with a set of pixels corresponding to an object from a separate image. If the two images used to create the composite image were captured under different lighting environments, an image forger may need to perform contrast enhancement on the pasted object so that lighting conditions match across the composite image. Failure to

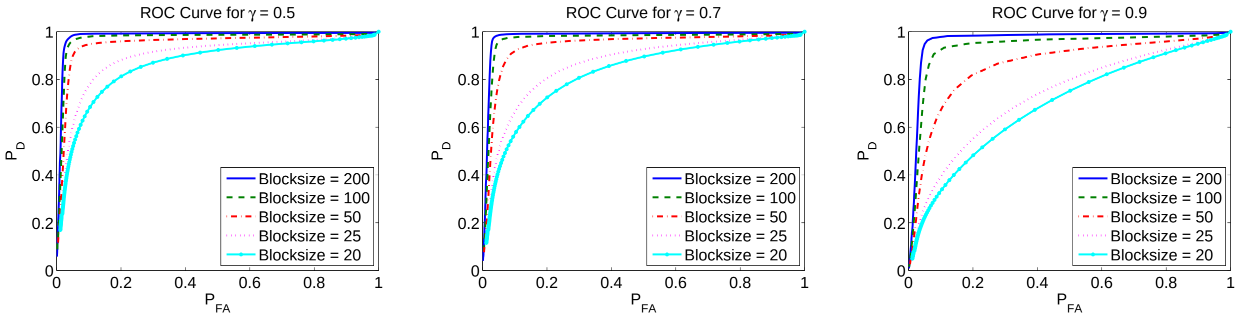
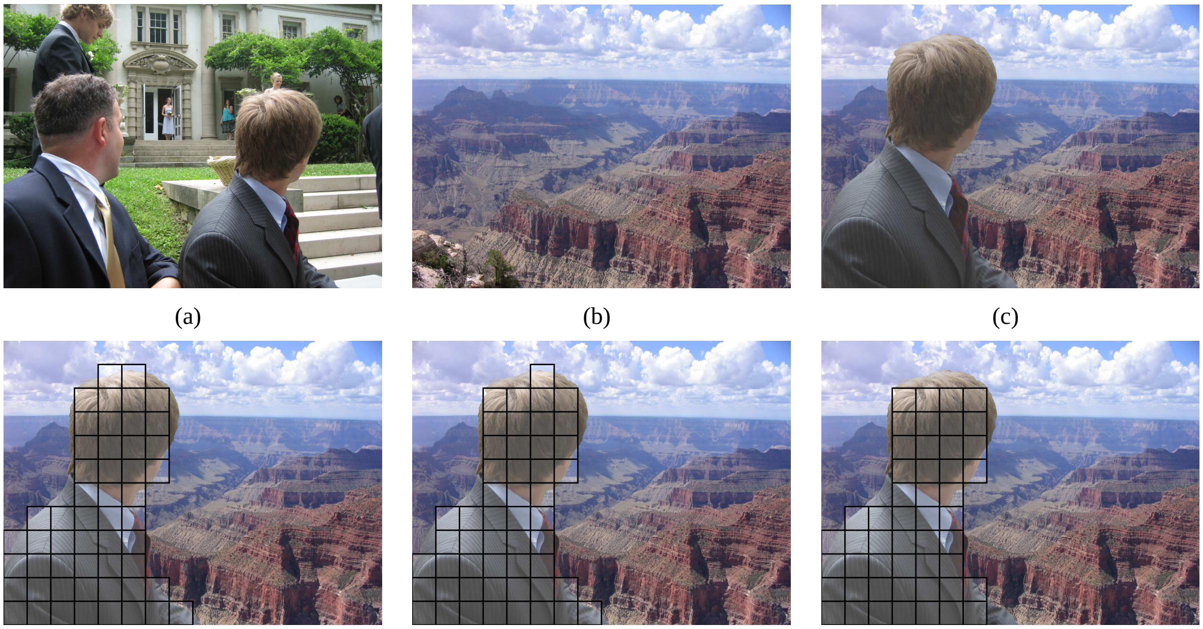


Fig. 4. ROC curves obtained using different testing block sizes for images altered by a power law transformation with $\gamma = 0.5$ (left), $\gamma = 0.7$ (center), and $\gamma = 0.9$ (right).



(a) (b) (c)

(d) (e) (f)

Fig. 5. Cut and paste forgery detection example showing (a) the unaltered image from which an object is cut, (b) the unaltered image into which the cut object is pasted, (c) the composite image, (d) red layer blockwise detections, (e) green layer blockwise detections, and (f) blue layer blockwise detections. Blocks detected as contrast enhanced are highlighted and boxed.

do this may result in a composite image which does not appear realistic. Image forgeries created in this manner can be identified by using localized contrast enhancement detection to locate the cut-and-pasted region. An example of a cut-and-paste image forgery in which the pasted region has undergone contrast enhancement is shown in Fig. 5 along with the localized contrast enhancement detection results obtained from our proposed forensic technique. Adobe Photoshop was used to create the forged image shown in 5(c) from the unaltered images shown in Figs. 5(a) and 5(b). Blocks corresponding to contrast enhancement detections are highlighted and outlined in black. In this example, each of these blocks contain pixels that correspond to the inauthentic object.

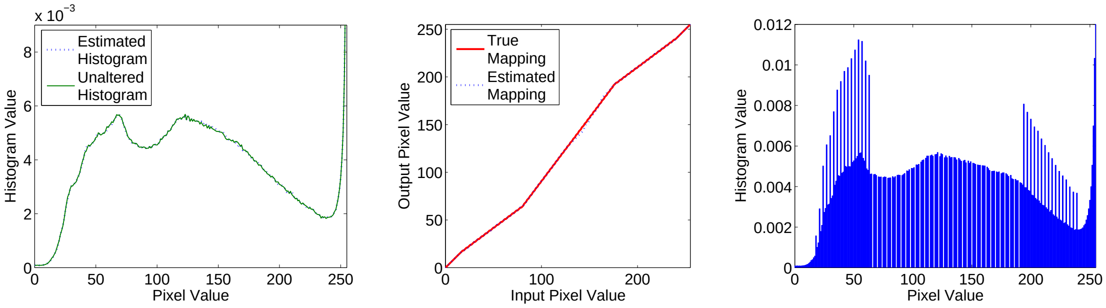


Fig. 6. The pixel value histogram of an unaltered image (left), the contrast enhancement mapping and its estimate (center), and the unaltered pixel value histogram and its estimate (right).

D. Forensic Contrast Enhancement Mapping Estimation

Once digital image manipulation has been identified, the next forensic task is to determine as much information as possible about the unaltered image and the operation used to modify it. In the case of images exhibiting evidence of contrast enhancement, we have developed an iterative technique to jointly estimate the contrast enhancement mapping used to modify an image as well as the images pixel value histogram before contrast enhancement. This technique operates by identifying an image's pixel value histogram entries most likely to correspond to contrast enhancement fingerprints, using these fingerprints to estimate the contrast enhancement mapping, obtaining an estimate of the unaltered pixel value histogram, then iteratively refining each estimate. Figure 6 shows an example of a contrast enhanced image's pixel value histogram, as well a comparison of our algorithm's estimate of the original histogram and contrast enhancement mapping to the true ones. As can be seen, we are able to achieve a highly accurate estimate of the contrast enhancement mapping and the unaltered pixel value histogram.

E. Detection of Additive Noise in Previously JPEG Compressed Images

When creating a digital image forgery, noise may be added to an image's pixel values to disguise visual traces of image forgery or an in attempt to mask statistical fingerprints left behind by other image altering operations. Previous work has dealt with the detection of noise added to specific regions of an image by searching for fluctuations in localized estimates of an image's signal to noise ratio (SNR) [7]. This method fails, however, when noise has been globally added to an image because this will not result in localized SNR variations. We have developed a technique capable detecting additive noise in previously JPEG compressed images by applying a pixel value mapping to the image, then searching of the mapping's intrinsic fingerprint. The mapping is designed in such a way that if noise is not present, the mapping's fingerprint will take the form of a periodic modulating signal within an image's pixel value histogram. If noise has been added to the image, this periodic fingerprint will be absent. Because of the fingerprint's periodic nature, we use a frequency domain representation of the transformed pixel value histogram, where the periodic signal takes the form of an spike centered at the fingerprint's fundamental frequency. This effect can be clearly seen in Figure 7.

To evaluate the performance of our additive noise detection technique, we compiled a set of 277 unaltered images taken by four different digital cameras from unique manufacturers. These images capture a variety of different scenes and were saved as JPEG compressed images using each camera's default settings. A set of altered images was created by decompressing each image and independently adding unit variance Gaussian noise to each pixel value. These altered images were then saved as bitmaps, along with decompressed versions of the original images, creating a testing database of 554 images. Next we used our additive noise detection test to determine if noise had been added to each image in the database. Detection and false

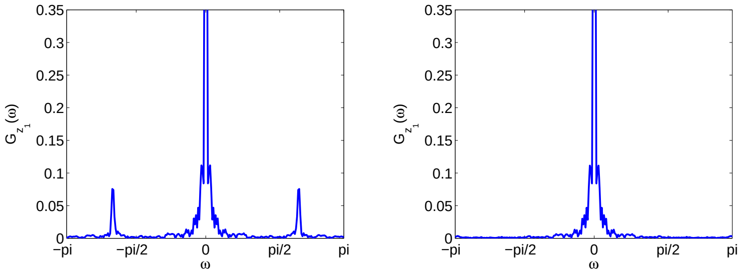


Fig. 7. Example showing the frequency domain representation of the transformed pixel value histogram from an unaltered image (left) as well as an altered version of the image to which unit variance Gaussian noise has been added (top right).

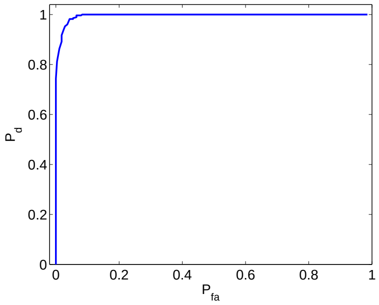


Fig. 8. Additive noise detection ROC curve for images which were JPEG compressed using default camera settings then altered by adding unit variance Gaussian additive noise.

alarm probabilities were determined at a series of decision thresholds and used to create an ROC curve showing the performance of our additive noise detection algorithm. This ROC curve is displayed in Fig. 8. A P_d of approximately 80% was achieved at a false alarm rate less than 0.4%. When the P_{fa} was held less than 6.5%, the P_d increased to nearly 99%. These results indicate that our detection scheme is able to reliably detect additive noise in images previously JPEG compressed using a camera's default settings.

III. Anti-Forensics of Digital Image Compression

In this section we discuss our work on anti-forensically removing compression fingerprints from digital images. We have developed a generalized framework to remove image compression fingerprints from transform coders [8] and shown how this framework can be adapted to remove JPEG compression fingerprints [8], [9] and DWT-based compression fingerprints [8], [10]. Additionally, we have developed a technique to remove blocking artifacts left by transform coders and shown how image compression anti-forensics can be used to make undetectable image forgeries [8], [11].

A. JPEG Compression Anti-Forensics

When a digital image is stored using JPEG compression, it is first segmented into 8×8 pixel blocks, then the DCT of each block is performed. Each block of DCT coefficients is quantized, then reordered into a single bitstream which is losslessly compressed. During decompression, each step in the process is inverted with the exception of quantization. Because quantization is not invertible, dequantization is performed by multiplying each quantized coefficient by the quantization step size. This process causes the DCT coefficients of the decompressed image to be clustered around integer multiples of the quantization

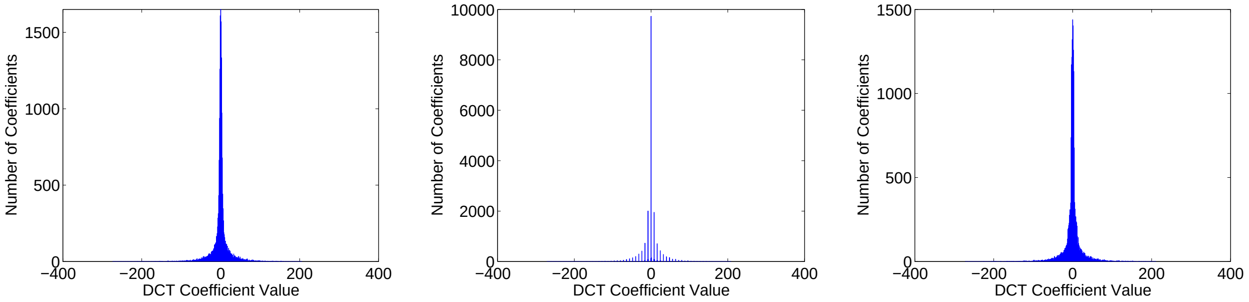


Fig. 9. Histogram of (2,2) DCT coefficients taken from an uncompressed version of the image shown in Fig. 10 (left), the same image after JPEG compression (center), and an anti-forensically modified copy of the JPEG compressed image(right).

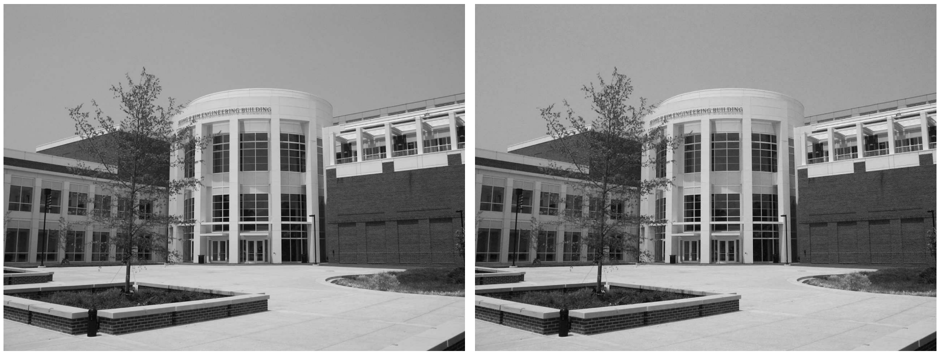


Fig. 10. Left: JPEG compressed image using a quality factor of 65. Right: Anti-forensically modified image.

step size as can be seen in Figure 9. These quantization artifacts correspond to the intrinsic fingerprints of JPEG compression which are used by several existing image forensic algorithms.

We have proposed an anti-forensic technique designed to remove JPEG compression fingerprints. It operates by first obtaining an estimate of the unquantized DCT coefficient distribution from the quantized DCT coefficients. Next, anti-forensic dither is added to the quantized DCT coefficients to remove quantization artifacts. The anti-forensic dither distribution is chosen based on the estimated unquantized DCT coefficient distribution. Figure 9 shows an example of a histogram of anti-forensically modified DCT coefficients which contain no JPEG compression fingerprints. Furthermore, this technique introduces very little distortion into the anti-forensically modified image, which can be seen in Figure 10 which shows a JPEG compressed image along with the same image after anti-forensic dither has been added to its DCT coefficients.

To test the effectiveness of our anti-forensic operation on a larger scale, we compressed then anti-forensically modified a set of 1338 images taken from the Uncompressed Colour Image Database [12]. These images were compressed using quality factors of 90, 70, and 50. After each image was anti-forensically modified, we used the algorithm described in [13] to estimate the quantization table used during compression and classify each image as never-compressed or previously JPEG compressed. Images were only classified as never-compressed if every quantization table entry was estimated as one or if no estimate could be obtained. We should note that performing classification in this manner significantly biases the output towards deciding that an image was previously JPEG compressed. Despite this, the classifier was unable to detect previous JPEG compression in 100% of the anti-forensically modified images.

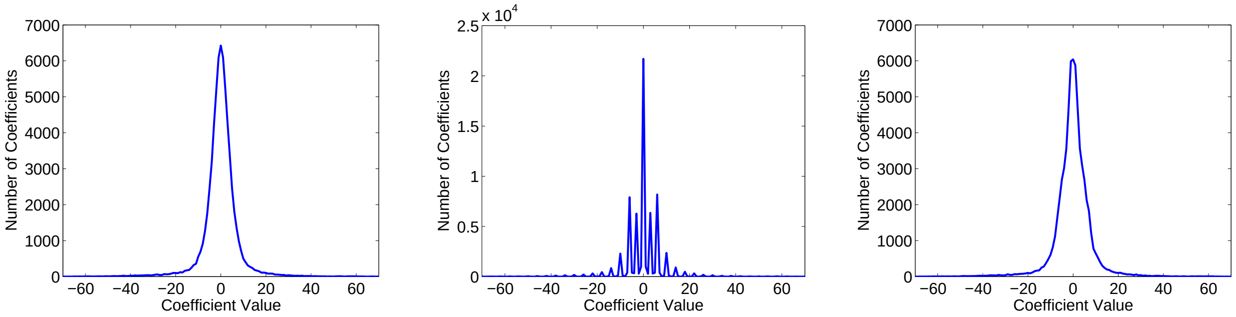


Fig. 11. Histogram of wavelet coefficients from the fourth level *HH* subband of a four level wavelet decomposition of the image shown in Fig. 12 (left), the same image after SPIHT compression (center), and the compressed image after anti-forensic dither has been applied (right).

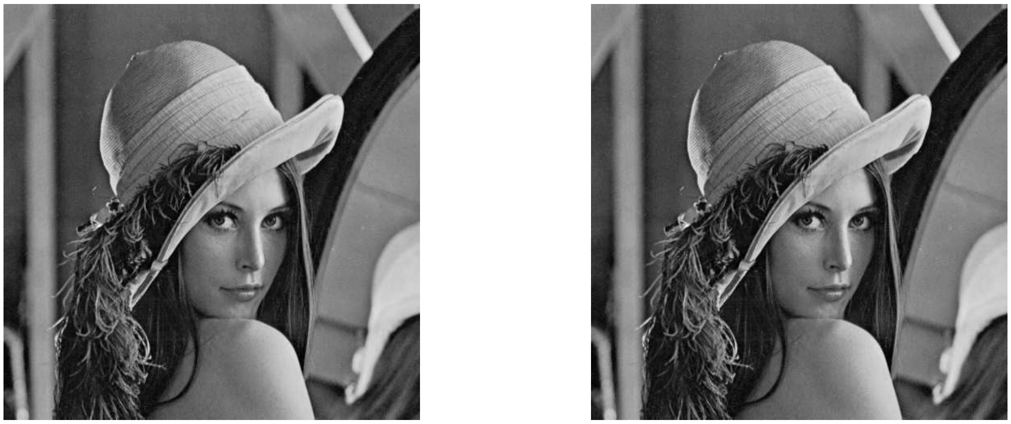


Fig. 12. Left: An image compressed using the SPIHT algorithm at a bit rate of 3 bits per pixel before the use of entropy coding. Right: The same image after anti-forensic dither has been applied to its wavelet coefficients.

B. Wavelet-Based Image Compression Anti-Forensics

Wavelet-based image compression leaves behind forensically significant intrinsic fingerprints in a similar manner to JPEG compression. When an image undergoes wavelet-based image compression, its discrete wavelet transform (DWT) is first computed, resulting in several subbands of wavelet coefficients. A tree structure is built out of the wavelet coefficients such that the least significant bits of each coefficient occur at the end of each branch. Compression is achieved by arranging this tree structure into a single bitstream, then truncating it so that only a fixed number of bits are retained. This has the same effect as applying quantization to each DWT subband and causes the compressed wavelet coefficients to cluster around a small set of quantized values as shown in Fig. 11. Existing forensic schemes use these artifacts to test for evidence of prior wavelet-based compression within images [14].

We have designed an anti-forensic technique designed to remove wavelet compression artifacts from previously compressed images. Our technique is similar in nature to the one which we have proposed to remove DCT quantization artifacts from JPEG compressed images. For each DWT subband, we first use the compressed wavelet coefficients to estimate the distribution of wavelet coefficients before compression. We then add anti-forensic dither to each wavelet coefficient, where the anti-forensic dither distribution is chosen using these estimates. Results indicate that our method is capable of removing compression artifacts from each wavelet subband's histogram of coefficients as can be seen in Fig. 11. Fig. 12 shows

Quality Factor	Proposed Method			Liew & Yan [15]	Zhai *et al.* [16]
	$s = 3$, $\sigma^2 = 3$	$s = 3$, $\sigma^2 = 2$	$s = 2$, $\sigma^2 = 2$		
90	0.0%	0.0%	0.0%	70.1%	99.6%
70	0.0%	0.0%	14.8%	99.2%	99.6%
50	0.0%	0.9%	62.7%	98.8%	99.6%
30	3.3%	23.0%	93.4%	99.6%	98.8%
10	97.9%	97.9%	100.0%	100.0%	82.8%

TABLE I
BLOCKING ARTIFACT DETECTION RESULTS.

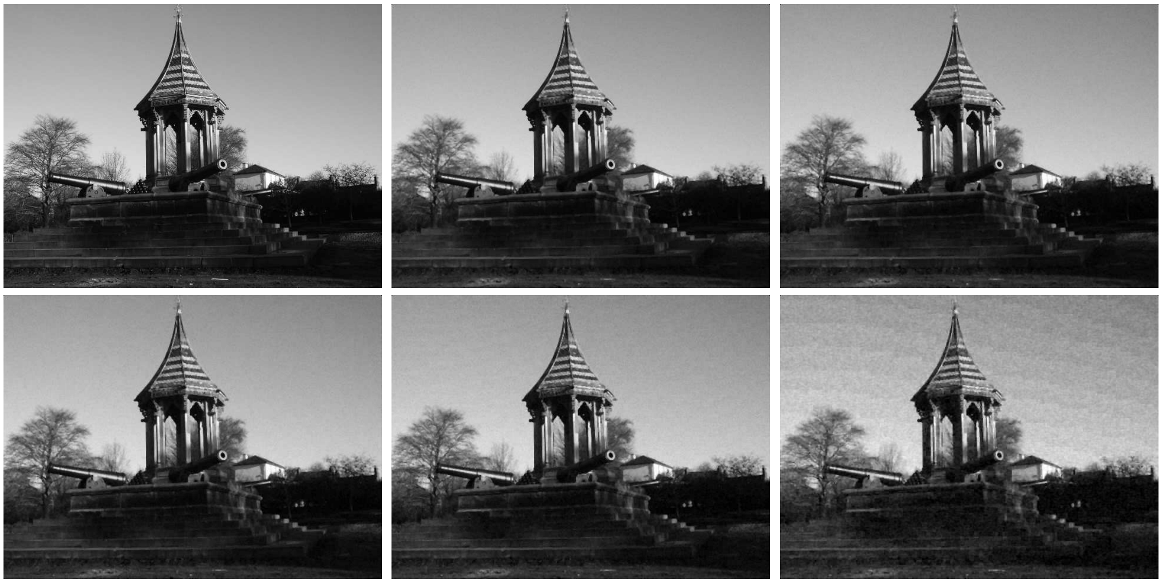


Fig. 13. Results of the proposed anti-forensic deblocking algorithm applied to a typical image (top left) after it has been JPEG compressed using a quality factor of 90 (top center), 70 (top left), 50 (bottom left), 30 (bottom center), and 10 (bottom right) followed by the addition of anti-forensic dither to its DCT coefficients.

that very little visual distortion is introducied into an anti-forensically modified image. Additionally, we have performed a larger scale test in which we compressed the 1338 images in the Uncompressed Colour Image Database using the SPIHT algorithm and used the forensic detector developed by Lin *et al.* to test for evidence of compression. In this test, we were able to fool the forensic algorithm into classifying an image as never-compressed 99.8% of the time.

If a previously JPEG compressed image is to be passed off as never having undergone compression, JPEG blocking artifacts must be removed from the image after anti-forensic dither has been applied to its DCT coefficients. Though a number of deblocking algorithms have been proposed since the introduction of the JPEG compression standard, the majority of these are ill suited for anti-forensic purposes. In order for an anti-forensic deblocking operation to be successful, it must remove all visual and statistical traces of block artifacts without resulting in forensically detectable changes to an image's DCT coefficient histograms. By contrast, existing deblocking algorithms are designed to only remove visible traces of blocking artifacts, particularly in heavily compressed images, and do not give consideration to the forensic detectability of compression artifacts in their output images. We propose an anti-forensic technique that removes statistical traces of JPEG blocking artifacts from an image to which anti-forensic dither has already been added. This is accomplished by first median filtering the image then adding Gaussian white noise. Both the support of the median filter s and the variance of the noise σ^2 are chosen based on the strength of the blocking artifacts.

To demonstrate the effectiveness of this anti-forensic deblocking operation as well as to illustrate its advantages over several existing deblocking algorithms, we have tested its ability to deceive the the forensic JPEG blocking artifact detector proposed in [13] along with the deblocking algorithms recently proposed in [15] and [16]. To do so, we compressed then deblocked each of the 244 images in the Uncompressed Colour Image Database [12]. Table I shows JPEG blocking artifact detection results obtained from our tests. These results clearly demonstrate that when the parameters s and σ^2 are chosen properly, our proposed algorithm is capable of removing statistical traces of blocking artifacts from images previously JPEG compressed at quality factors of 30 and above. Furthermore, these results indicate that while the algorithms presented in [15] and [16] are able to remove visual traces of blocking artifacts, they do not entirely remove all statistical traces and are not appropriate for anti-forensic purposes. A visual comparison of images deblocked using our proposed technique suggests that compression artifacts can be removed from images previously compressed using quality factors of 50 or higher without introducing significant visual distortion.

C. Undetectable Image Tampering Using Anti-Forensics

We have demonstrated that our DCT compression artifact removal technique and our proposed anti-forensic deblocking technique can be used to fool a variety of image forensic algorithms. Techniques have been proposed to detect a second application of JPEG compression to an image previously JPEG compressed [7], [17]. By applying our anti-forensic techniques before recompression, we are able to prevent the occurance of double compression fingerprints. Because most digital cameras make use of proprietary quantization tables, an image's compression history can be used to help identify the camera used to capture it [18]. We are able to wipe away an image's compression history using our anti-forensic techniques and insert a fake one. This allows us to falsify the originating camera of a digital image. Furthermore, techniques have been proposed to identify cut-and-paste image forgeries by detecting spatially localized discrepancies in an image's JPEG compression signature [19], [20]. We have demonstrated that our proposed anti-forensic operations can be used to successfully remove the fingerprints that each of these techniques rely on [8], [11].

IV. Video Frame Deletion Forensic and Anti-Forensics

To verify the authenticity of digital video files, digital forensic techniques have been developed to detect video manipulation and identify digital video forgeries. Of particular importance is the detection of video frame deletion or addition and recompression. Frame deletion may be performed by a video forger who wishes to remove certain portions of a video sequence such as a person's presence in a surveillance video.

In prior work, Wang and Farid demonstrated that frame deletion or insertion followed by recompression introduces a forensically detectable fingerprint into MPEG video [21]. Their work, however, relies on human inspection of the P-frame prediction error sequence to detect frame deletion and cannot be applied to newer video coders that used variable length group of picture (GOP) sequences. We have developed a new theoretical model of video frame deletion fingerprints. We have used this to create new automatic frame deletion detection techniques that do not rely on human inspection and are suitable for use with newer video coders that use variable GOP lengths [22]. Additionally, we developed an anti-forensic technique capable of removing frame deletion fingerprints from a digital video [22], [23]. Furthermore, we used our knowledge of how a forger is able to anti-forensically modify a video to create a technique to detect the use of frame deletion anti-forensics [22], [23].

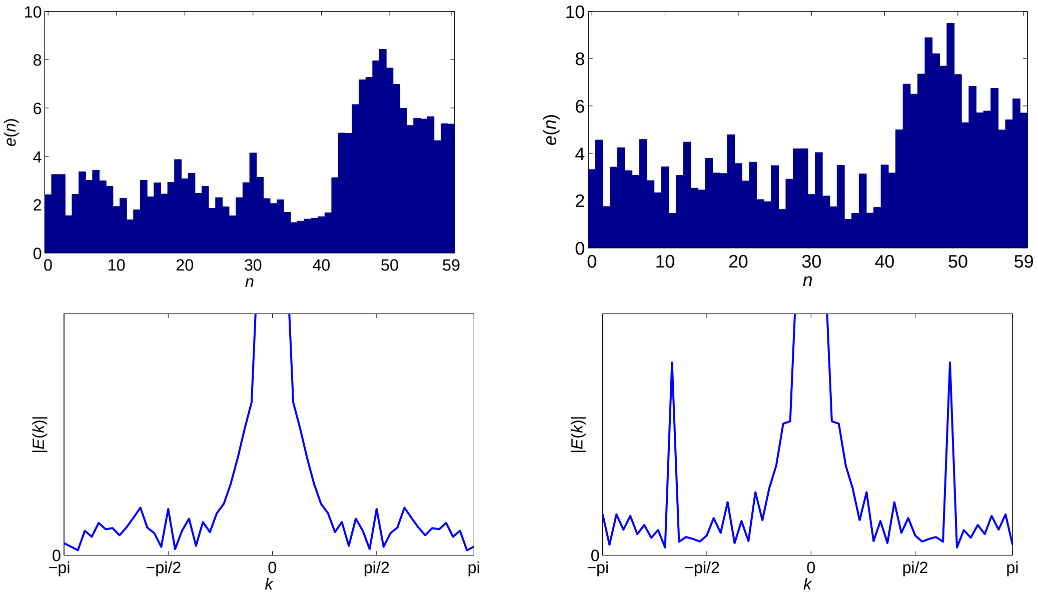


Fig. 14. P-frame prediction error sequence (top left) and the magnitude of its DFT (bottom left) obtained from an unedited, MPEG compressed version of the 'Carphone' video sequence along with the P-frame prediction error sequence (top right) and the magnitude of its DFT (bottom right) obtained from the same video after frame deletion followed by recompression.

A. Frame Deletion Detection

Due to the size of uncompressed digital video files, virtually all digital video undergoes compression during storage or transmission. Video encoders exploit redundancy between frames by predicting certain frames from others, then storing the prediction error. To prevent error propagation, the video sequence is divided into segments, where each segment is referred to as a group of pictures (GOP), during MPEG video compression. When frames are deleted from a digital video, the sequence of frames is shifted. During recompression, frames from different initial GOPs will be grouped together in each new GOP. This causes an increase in the prediction error for P-frames predicted across old GOPs. These spikes in the sequence of P-frame prediction errors $e(n)$, which can be seen in Fig. 14, are used as frame deletion fingerprints.

If the video coder used to compress the video uses fixed length GOP sequences, we have demonstrated that frame deletion fingerprints have the following properties

Property 1: The temporal fingerprint's repetitive pattern corresponds to a disproportionate increase in $e(n)$ exactly once per fingerprint period.

Property 2: The period T of the temporal fingerprint is equal to the number of P-frames within a GOP.

Property 3: Define the phase ϕ of the temporal fingerprint as the number of P-frames within a GOP before the increase in $e(n)$ due to frame deletion. The phase is determined by the equation $\phi = \lfloor |\mathcal{A}|/n_P \rfloor$, where n_P is the number of P-frames within a GOP, $\mathcal{A}$ is the set of frames at the beginning of each GOP that belonged to the same GOP during the initial application of compression, $|\mathcal{A}|$ denotes the cardinality of $\mathcal{A}$, and $\lfloor \cdot \rfloor$ denotes the floor operation.

We have used these properties to create a mathematical model of frame deletion fingerprints. This model identifies the period of frame deletion fingerprints and location of a peak in the DFT of the prediction error sequence cause by frame deletion fingerprint. We then formulated frame deletion as a hypothesis testing problem and used our model to create an automatic frame deletion detection technique suitable for videos compressed using fixed length GOPs. Additionally, we constructed a model of frame deletion fingerprints when the video coder does not use a fixed length GOP. In this case, frame deletion fingerprints are not

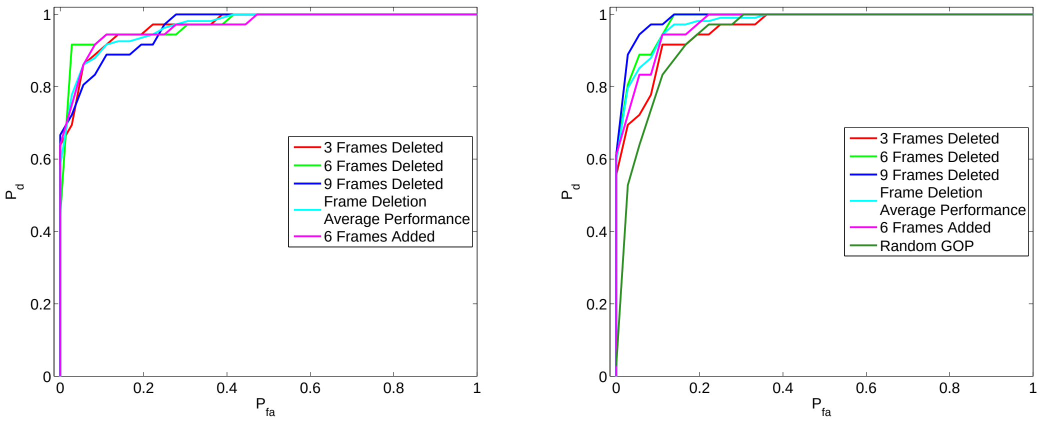


Fig. 15. ROC curves for our frame deletion detector designed to operate on videos with fixed length GOPs (left) and variable length GOPs (right) obtained by testing against different amounts frame deletion and addition.

periodic. Using this model, we constructed a frame deletion detection technique capable of operation on videos compressed by modern coders that allow for variable GOP lengths.

To test the forensic effectiveness of our proposed frame deletion detectors, we first created a database of forged videos. To do this, we deleted 3, 6, and 9 frames from the beginning of each of 36 unaltered standard video sequences compressed using a fixed length GOP, then recompressed each video. This corresponded to removing 1/4, 1/2, and 3/4 of a GOP respectively. To test against frame addition, we added 6 frames to the beginning of each unaltered video sequence compressed with a fixed length GOP, then recompressed these videos. Additionally, we deleted 6 frames from the videos compressed using randomly varying GOP lengths. We then used each of our proposed detection techniques in conjunction with a series of different decision thresholds to determine if frame deletion or addition had occurred in each video. The results of these tests were used to create the ROC curves for each detector shown in Fig. 15. We can see from these ROC curves that both detectors' performance remains consistent regardless of the number of frames deleted. Furthermore, we can see that both detectors were able to achieve an average P_d of at least 85% at a false alarm rate less than 5%. Both detectors also achieved a P_d of at least 90% at a false alarm rate less than 10%. These results indicate that both detectors can be used to reliably detect frame deletion.

B. Frame Deletion Anti-Forensics

If a forger wishes to undetectably delete a sequence of frames from a digital video, they must ensure that frame deletion fingerprints do not occur in the videos P-frame prediction error sequence. We have developed an anti-forensic technique to prevent these fingerprints from occurring. Our anti-forensic operation works by modifying the encoding process so that the P-frame prediction error sequence matches a target prediction error sequence that does not contain the temporal fingerprint. The value of $e(n)$ is increased to the target value $\hat{e}(n)$ for a given P-frame by changing the frame's predicted value in a manner that increases the prediction error. Since the anti-forensically modified video must be capable of being decompressed by a standard MPEG decoder, we accomplish this modifying the motion vectors of each frame's macroblocks in order to increase the prediction error. After this is done, new prediction error values are obtained and stored for each macroblock whose motion vectors are modified. We note that though the prediction error is increased for an anti-forensically modified P-frame, the decompressed P-frame remains essentially unchanged by anti-forensic modification because the new prediction error is stored during compression, then added back to the new predicted frame during decompression.

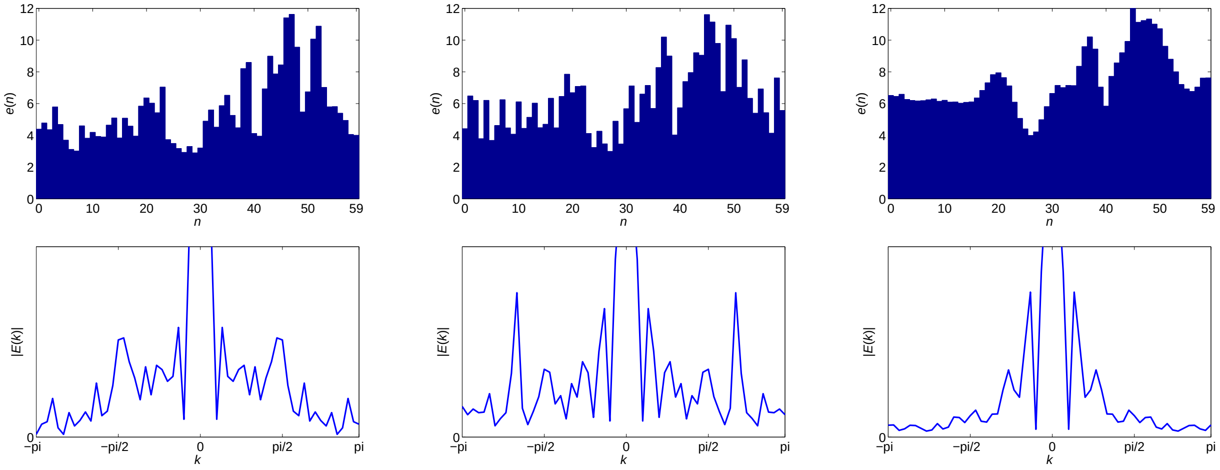


Fig. 16. P-frame prediction error sequences (top row) and the magnitudes of their respective DFTs (bottom row) obtained from an untampered MPEG compressed version of the 'Foreman' video (left column), as well as from the same video after the first six frames were deleted followed by recompression without anti-forensic modification (middle column) and with the use of our proposed anti-forensic technique (right column).

To evaluate the performance of our proposed frame deletion anti-forensic technique, we deleted six frames from each unaltered video compressed using a fixed GOP structure, then recompressed each video while applying our anti-forensic technique. An example of typical results achieved by our proposed anti-forensic technique is shown in Fig. 16. This figure displays the P-frame prediction error sequence taken from an untampered MPEG compressed version of the 'Foreman' video, as well as the P-frame prediction error sequences obtained after deleting the first six frames then recompressing the video with and without applying our anti-forensic temporal fingerprint removal technique. Frame deletion fingerprints features prominently in the prediction error sequence of the video in which frames are deleted without the use of our anti-forensic technique, particularly in the frequency domain. By contrast, these fingerprints are absent from the prediction error sequence when our anti-forensic technique is used to hide evidence of frame deletion.

Additionally, we examined the ability of our proposed anti-forensic technique to fool each of our automatic frame deletion detection techniques. To do this, we used both of our proposed detection techniques to classify each video in our databases of 36 unaltered and 36 anti-forensically modified videos as unaltered or one from which frames had been deleted. We used this data to generate a new set of ROC curves for each of our frame deletion detection techniques when frame deletion has been disguised using anti-forensics. These ROC curves are displayed in Fig. IV-B. In this figure, the dashed line represents the performance of a decision rule that randomly classifies a video as forged with a probability equal to P_{fa}. Reducing a detection technique's performance to this level corresponds to making it equivalent to a random guess. As we can see from Fig. IV-B, both frame deletion detection techniques perform at or near this level when our anti-forensic technique is applied to a video.

C. Detecting the Use of Frame Deletion Anti-Forensics

In order to remove frame deletion fingerprints from the P-frame prediction sequence of a video, that video's motion vectors must be altered in order to increase the prediction error. Despite this, the true motion present in the video does not change. As a result, there is a discrepancy between many of the motion vectors stored in an anti-forensically modified video and the true motion of that video scene. This is not the case for an unaltered video because normal video encoders will attempt to estimate scene

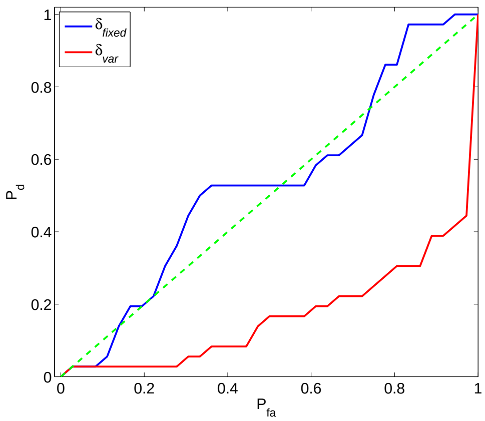


Fig. 17. Experimental results showing ROC curves for our fixed GOP frame deletion detector δ_{fixed} and our variable length GOP frame deletion detector δ_{var} obtained by testing on anti-forensically modified videos.

motion as accurately as possible in order to minimize each frame's prediction error. Accordingly, these discrepancies between a video's stored motion vectors and the actual motion of the scene are fingerprints left by frame deletion anti-forensics. We have designed a technique to detect the use of frame deletion anti-forensics. Our detection technique operates by comparing a compressed video's P-frame motion vectors to an estimate of the true motion present in the video scene. This is done by first decompressing the video in question, then performing motion estimation on the video to obtain a new set of row and column motion vectors.

In order to evaluate the performance of our technique designed to detect the use of frame deletion anti-forensics, we re-examined the videos in our database of 36 unaltered and 36 anti-forensically modified MPEG-2 compressed videos. We used our proposed detector to classify each video as unmodified or anti-forensically modified , then used these results to generate the ROC curve shown in Fig. 18. The results of this experiment show that our proposed detector achieved perfect detection (i.e. a P_d of 100% at a P_{fa} of 0%). These results are slightly misleading, however, because the motion vectors of the videos in the unaltered database are obtained using an exhaustive search. In reality, many video encoders use efficient algorithms to peform motion estimation. To evaluate the performance of our proposed frame deletion anti-forensics detection technique under less favorable conditions, we repeated the previous experiment using the three step search algorithm proposed by Zhu and Ma [24] during compression.

We can see from Fig. 18 that the performance of our proposed detector is degraded in this scenario. While the detection of frame deletion anti-forensics can still be performed, it must be done with a higher false alarm rate. This suggests that if a forensic investigator's maximum acceptable false alarm rate is sufficiently low, a video forger using anti-forensics is likely to avoid detection. To mitigate this, a forensic investigator may wish to repeat frame deletion anti-forensics detection using a decision threshold corresponding to a higher false alarm rate, but not immediately assume that detections correspond to forged videos. Instead, these videos can be flagged for closer investigation using additional forensic techniques.

V. Evaluation of Anti-Forensics and the Trade-off Between Forensics and Anti-Forensics

In the past, the performance of digital forensic techniques has been measured using traditional tools from decision theory. While these tools can adequately evaluate forensic techniques, they often are poorly suited to measure the performance of anti-forensic operations. For example, should a missed forgery detection in

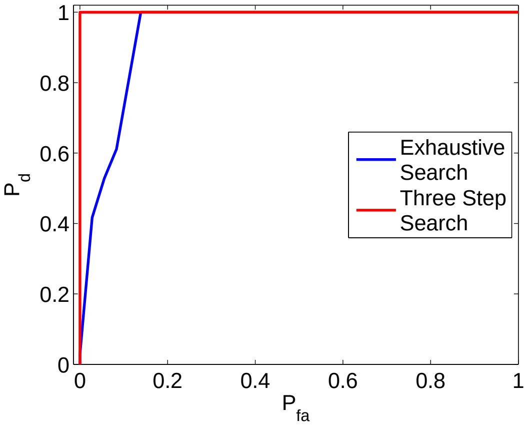


Fig. 18. ROC curves for the anti-forensics detector δ_{mv} when tested on video data compressed using an exhaustive search to determine motion vectors and video data encoded using a three step motion vector search algorithm.

an anti-forensically modified file be counted the same as one in which the file was not anti-forensically modified? If an anti-forensic operation is able to successfully remove fingerprints left by a particular forgery operation but introduces new fingerprints of its own, how do we evaluate its effectiveness?

We have addressed these problems by developing a set of techniques to evaluate the performance of anti-forensic operations [22], [25]. Additionally, we constructed a game theoretic framework to evaluate the dynamics between a forger and a forensic investigator [22], [25]. This framework can be used to determine the probability that a forgery will be detected when both a forger and forensic investigator are using optimal anti-forensic and forensic detection strategies.

A. Performance Analysis of Anti-Forensics

To properly evaluate the performance of an anti-forensic technique, we have developed a new measure known as the *anti-forensic susceptibility* of a forensic technique to anti-forensics. This measure avoids unintentional bias towards overestimating an anti-forensic operation's performance by counting only missed forensic detections caused by anti-forensics.

The anti-forensic susceptibility is a measure between 0 and 1 of the decrease in effectiveness of a forensic detector caused by the use of an anti-forensic operation. It is defined as decrease in a forensic detection technique's probability of detection caused by the use of anti-forensics divided by the maximum decrease in the forensic detection techniques's probability of detection that an anti-forensic operation needs to cause in order to render forensics ineffective. This corresponds to the ratio A/B in Fig. 19.

We measured the anti-forensic susceptibility to measure the performance of our video frame deletion anti-forensic technique discussed in Section IV-B. These results are displayed in Fig. 20. These results show that for all $P_{fa} \leq 80\%$, our anti-forensic technique acheived an anti-forensic susceptibility of .7 or greater. Furthermore, for all $P_{fa} \leq 20\%$, the frame deletion detector performs no better than a random decision if anti-forensics is used.

B. Game Theoretic Analysis of the Trade-off Between Forensics and Anti-Forensics

A forger may choose to reduce the strength of fingerprints left by their anti-forensic operation by decreasing the strength at which they apply anti-forensics. They must be careful, however, because this will cause a corresponding increase in the strength of the manipulation fingerprints that remain after anti-forensics has been used. The forensic investigator, meanwhile, must ensure that the combination of the

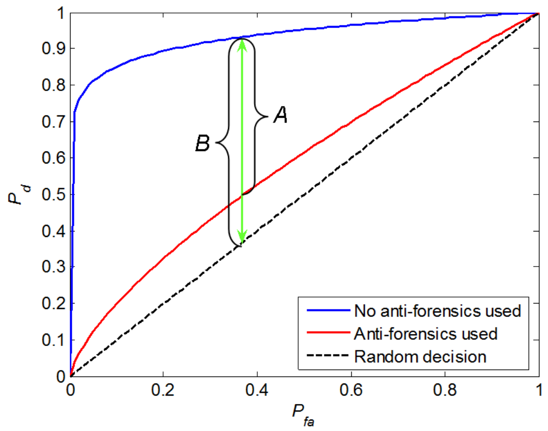


Fig. 19. Example relating the anti-forensic effectiveness of an anti-forensic operation to the ROC curves achieved by a forensic technique when anti-forensics is and is not used. The anti-forensic effectiveness at a given false alarm level is the ratio A/B.

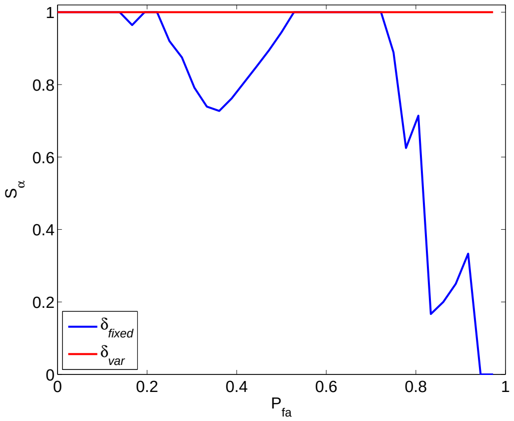


Fig. 20. Experimental results showing anti-forensic susceptibility plots for frame deletion detection using our fixed GOP detector δ_{fixed} and variable GOP length detectorδ_{var} obtained by testing on anti-forensically modified videos.

false alarm rates from their techniques to detect editing and the use of anti-forensics is below a constant false alarm rate. As a result, the forger and forensic investigator must both balance a set of trade-offs that depend upon the actions of the other party.

We have developed a game theoretic framework to evaluate the interplay between a forger and a forensic investigator. In this framework we define the utility of the forensic investigator as the probability that they will detect either forgery fingerprints or fingerprints left by the use of anti-forensics. The utility of the forger is negative one times the utility of the forensic investigator minus a penalty term for perceptual distortion introduced into the forgery by the use of anti-forensics. These utility functions can be used to identify the Nash equilibrium strategy of both the forger and forensic investigator. If one player operates at their Nash equilibrium strategy, the other player gains no advantage by choosing any other strategy, thus both players have no incentive to deviate from the Nash equilibrium strategies. If no closed for expression for these utilities exist, the Nash equilibria can be determined numerically. Furthermore, by determining the probability of forgery detection at the Nash equilibrium for each total false alarm level between zero and one, a new ROC curve can be constructed showing the forensic investigator's ability to detect forgeries if both players act rationally. We define this ROC curve as the *Nash equilibrium receiver*

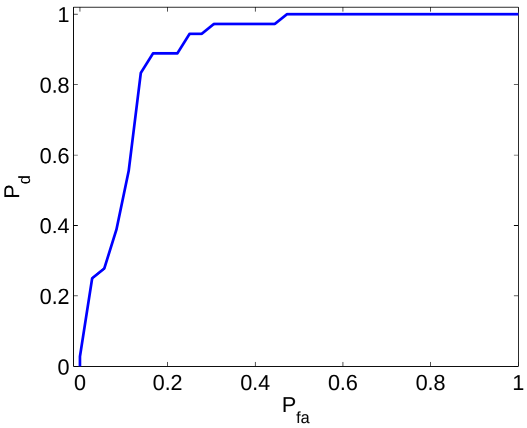


Fig. 21. Nash equilibrium ROC curve for video frame deletion detection.

operating characteristic curve, or NE ROC curve.

We used our game theoretic framework to determine the probability of forgery detection at Nash equilibrium for the problem of video frame deletion. To do this, we modified our anti-forensic technique to operate at variable strengths by making the anti-forensic increase in each P-frame's prediction error adjustable. We then modified each video with several different anti-forensic strengths and performed frame deletion and anti-forensics detection as before. This allowed us to numerically identify the Nash equilibrium strategies for a range of constraints on the forensic investigator's total false alarm rate between 0% and 100%. We used these results to create the NE ROC curve displayed in Fig. 21. From this curve we can see that if the forensic investigator must operate with a total probability of false alarm constraint of 10% or less, frame deletion forgeries are difficult to detect. If the forensic examiner is able to relax their probability of false alarm constraint to roughly 15% or greater, then they will be able to detect frame deletion forgeries at a rate of at least 85%.

References

[1] M. Chen, J. Fridrich, M. Goljan, and J. Lukáš, "Determining image origin and integrity using sensor noise," *IEEE Trans. on Inform. Forensics Security*, vol. 3, no. 1, pp. 74–90, March 2008.

[2] M. Stamm and K.J.R. Liu, "Blind forensics of contrast enhancement in digital images," in *Proc. ICIP*, San Diego, CA, USA, Oct. 2008, pp. 3112–3115.

[3] M.C. Stamm and K.J.R. Liu, "Forensic detection of image tampering using intrinsic statistical fingerprints in histograms," in *Proc. APSIPA Annual Summit and Conference*, Oct. 2009.

[4] M. C. Stamm and K. J. Ray Liu, "Digital image source coder forensics via intrinsic fingerprints," *submitted to IEEE Trans. Information Forensics and Security*, 2010.

[5] M. C. Stamm and K.J.R. Liu, "Forensic estimation and reconstruction of a contrast enhancement mapping," in *Proc. ICASSP*, Mar. 2010.

[6] G.E. Healey and R. Kondepudy, "Radiometric CCD camera calibration and noise estimation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 3, pp. 267–276, Mar. 1994.

[7] A.C. Popescu and H. Farid, "Statistical tools for digital forensics," in *6th International Workshop on Information Hiding*, Toronto, Canada, 2004.

[8] M. C. Stamm and K. J. R. Liu, "Anti-forensics of digital image compression," *IEEE Trans. Information Forensics and Security*, vol. 6, no. 3, pp. 1050 –1065, Sep. 2011.

[9] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K.J.R. Liu, "Anti-forensics of JPEG compression," in *Proc. ICASSP*, Mar. 2010.

[10] M. C. Stamm and K. J. R. Liu, "Wavelet-based image compression anti-forensics," in *Proc. IEEE Int. Conf. Image Processing*, Sep. 2010, pp. 1737 – 1740.

[11] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K. J. R. Liu, "Undetectable image tampering through JPEG compression anti-forensics," in *Proc. IEEE Int. Conf. Image Processing*, Sep. 2010, pp. 2109 – 2112.

[12] G. Schaefer and M. Stich, "UCID: an uncompressed color image database," in *Proc. SPIE: Storage and Retrieval Methods and Applications for Multimedia*, 2003, vol. 5307, pp. 472–480.

[13] Z. Fan and R. de Queiroz, “Identification of bitmap compression history: JPEG detection and quantizer estimation,” *IEEE Trans. Image Processing*, vol. 12, no. 2, pp. 230–235, Feb 2003.

[14] W. S. Lin, S. K. Tjoa, H. V. Zhao, and K. J. Ray Liu, “Digital image source coder forensics via intrinsic fingerprints,” *IEEE Trans. Information Forensics and Security*, vol. 4, no. 3, pp. 460–475, Sept. 2009.

[15] G. Zhai, W. Zhang, X. Yang, W. Lin, and Y. Xu, “Efficient image deblocking based on postfiltering in shifted windows,” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 18, no. 1, pp. 122–126, Jan. 2008.

[16] A.W.-C. Liew and H. Yan, “Blocking artifacts suppression in block-coded images using overcomplete wavelet representation,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 14, no. 4, pp. 450–461, April 2004.

[17] T. Pevný and J. Fridrich, “Detection of double-compression in JPEG images for applications in steganography,” *IEEE Trans. on Information Forensics and Security*, vol. 3, no. 2, pp. 247–258, June 2008.

[18] H. Farid, “Digital image ballistics from JPEG quantization,” Tech. Rep. TR2006-583, Dept. of Computer Science, Dartmouth College, 2006.

[19] J. He, Z. Lin, L. Wang, and X. Tang, “Detecting doctored JPEG images via dct coefficient analysis,” in *Proc. of ECCV*, 2006, vol. 3593, pp. 423–435.

[20] S. Ye, Q.n Sun, and E.-C. Chang, “Detecting digital image forgeries by measuring inconsistencies of blocking artifact,” in *Proc. of ICME*, 2007, pp. 12–15.

[21] W. Wang and H. Farid, “Exposing digital forgeries in video by detecting double MPEG compression,” in *Proc. ACM Multimedia and Security Workshop*, Geneva, Switzerland, 2006, pp. 37–47.

[22] Lin W. S. Stamm, M. C. and K. J. R. Liu, “Temporal forensics and anti-forensics in digital videos,” *submitted to IEEE Trans. Information Forensics and Security*.

[23] M. C. Stamm and K. J. R. Liu, “Anti-forensics for frame deletion/addition in MPEG video,” in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Prague, Czech Republic, May 2011, pp. 1876 – 1879.

[24] S. Zhu and K.-K. Ma, “A new diamond search algorithm for fast block-matching motion estimation,” *IEEE Trans. Image Processing*, vol. 9, pp. 287–290, Feb. 2000.

[25] Lin W. S. Stamm, M. C. and K. J. R. Liu, “Forensics vs. anti-forensics: A decision and game theoretic framework,” in *to appear in Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Kyoto, Japan, Mar. 2012.